Substitution Estimators

SSE 708: Machine Learning in the Era of Big Data

David McCoy PhD, MSc
Division of Biostatistics, UC Berkeley
Edward's Lifesciences

July 7-11 2025

Table of Contents

What is a substitution estimator?

- If the parameter of interest is some algorithm (mapping), Ψ applied to the true data-generating distribution, P_0, so $\Psi(P_0)$ equals the quantity of interest, a substitution substitutes an estimate of P_0, say P_n^* and then uses the same mapping, or $\Psi(P_n^*)$.
- For example, consider a simple situation with $O = Y \sim P_0$, $\Psi(P_0) = E_0(Y) = \sum_y y * P_0(Y = y)$.
- Then, $\Psi(P_n) = \sum_y y * P_n(Y = y)$ is the substitution estimator.
- Assume the data are n independent observations of $Y_i, i = 1, ..., n$.
- The empirical distribution P_n just assigns probability $1/n$ to every observation, so the substitution estimator can be rewritten as:

$$\Psi(P_n) = \frac{1}{n}\sum_{i=1}^{n} Y_i = \bar{Y}$$

or just the sample average.

Example: Sample Variance

- Setup

$$\begin{aligned} Data &= Y_i, i = 1, ..., n, \\ \Psi(P_0) &= var_0(Y) = E_0(Y - E_0)^2 \\ &= \sum_y (y - E_0 Y)^2 P_0(Y = y) \end{aligned}$$

- We derived the substitution estimator for the mean $E_0(Y)$ is $\bar{Y}$
- Again, we plug in P_n for the unknown P_0 and get

$$\begin{aligned} \Psi(P_n) &= var_n(Y) = \\ &= \sum_y (y - \bar{Y})^2 P_n(Y = y) \\ &= \frac{1}{n} \sum_{i=1}^{n} (Y_i - \bar{Y})^2 \end{aligned}$$

Substitution Estimator for Average Treatment Effect (ATE), $\Psi(P_X) = (Y_1 - Y_0)$

David McCoy PhD, MSc
Division of Biostatistics, UC Berkeley
Edward's Lifesciences

[Robins(1999)] proposed substitution estimators (G-computation) for causally inspired estimands.

- Recall that under assumptions:

$$E(Y(1) - Y(0)) = E_{W,0}\{E_0(Y|A=1,W) - E_0(Y|A=0,W)\}$$

→ Data is: $O = (W, A, Y) \sim P_0 \in \mathcal{M}^{NP}$

→ Under assumptions discussed earlier, $\Psi(P_X) = \Psi(P_0) = \Psi(Q_0)$, where Q_0 represents both the distribution of $Y \mid W, A$ and distribution of W.

$$\Psi(Q_0) = E_{W,0}\{E_0(Y \mid A=1, W) - E_0(Y \mid A=0, W)\}$$

→ Let $Q_0(A, W) \equiv E_0(Y \mid A, W)$ and $Q_{0,W}(w) = P_0(W = w)$, then

$$\Psi(Q_0) = \sum_w \{Q_0(1, w) - Q_0(0, w)\} Q_{0,W}(w)$$

Substitution Estimator for ATE

David McCoy PhD, MSc
Division of Biostatistics, UC Berkeley
Edward's Lifesciences

- The estimand (parameter) of interest is:

$$\Psi(Q_0) = \sum_{w} \{Q_0(1, w) - Q_0(0, w)\} Q_{0,W}(w)$$

- The <u>Substitution Estimator</u>

$$\Psi(Q_n) = \frac{1}{n} \sum_{i=1}^{n} \{Q_n(1, W_i) - Q_n(0, W_i)\}$$

  → $Q_{n,W}(W_i) = 1/n$ (the empirical) and $Q_n(A, W)$ is a regression (or machine learning) regression of Y on (A, W).

- Provides a general approach for nonparametric estimation of parameters using machine learning.

Another Example: Counterfactual Mean Difference within Subgroups

- Data is like above ($O = (W, A, Y) \sim P_0 \in \mathcal{M}^{NP}$).
- Additionally define a variable V which is one of the W's, so $V \subset W$.
- Causal Parameter is $\Psi(P_X)(v) = E_X(Y_1 - Y_0 \mid V = v)$.
- Say V is categorical age, then the parameter above is the stratified average treatment effect, within strata of age $V = v$.
- Estimand With assumptions, then $\Psi(P_X)(v) =$

$$\Psi(P_0)(v) = E_0\{E_0(Y \mid A = 1, W) - E_0(Y \mid A = 0, W) \mid V = v\}$$

- Substitution Estimator

$$\Psi(Q_n)(v) = \frac{1}{n_v} \sum_{i=1}^{n_v} I(V_i = v)\{Q_n(1, W_i) - Q_n(0, W_i)\}$$

where n_v is the number of observations with $V_i = v$.

Understanding concepts in statistics from simulation

- One of the best methods to understand:
  1. how causal graphs are connected to data,
  2. understanding what the target estimand (parameter of interest)
  3. how to estimate from data the parameter of interest, and
  4. what statistical measures of uncertainty mean (the sampling distribution).
- We will now use this tool to solidify the understanding thus far of the topics we've been discussing (defining parameter of interest, data-generating mechanism, parameter of interest, estimation and inference).

Simulation that goes from the causal graph to simulated data

The Causal Model

$$\begin{tikzcd}
 & V \subset W \arrow[dl] \arrow[dr] & \\
A \arrow[rr] & & Y
\end{tikzcd}$$

The Data Generating Distribution

- $W_1 = \sim N(0, \sigma^2_W)$
- $V = W_2 = Uniform(0, 1, 2, 3, 4)$

- $log\{\frac{P(A=1|W)}{1-P(A=1|W)}\} = \alpha_0 + \alpha_1 * W_1 + \alpha_2 * V + \alpha_3 * W_1 * V = logit(g(W))$
  - → So, distribution of $A \mid W$ is a coin flip with probability of $A = 1$ being $g(W)$.
- $log\{\frac{P(Y=1|A,W)}{1-P(Y=1|A,W)}\} = \beta_0 + \beta_1 * W_1 + \beta_2 * V + \beta_3 * A + \beta_4 A * V = logit(Q(A, W))$
  - → Distribution of $Y \mid W, A$ is a coin flip with probability of $Y = 1$ being $Q(A, W)$.

R-code

```
## Sample size is n
set.seed(1231231)
n<-500
sigmaW<-0.5

## Generate random W and V
W1<-rnorm(n,0,sigmaW)
V<-sample(0:4,n,replace=T)

## Generate random A given W,V
alpha0<-0
alpha1<-1
alpha2<--1
alpha3<-0.5
PA.1givenWV<-1/(1+exp(-(alpha0+alpha1*W1+alpha2*V+alpha3*W1*V)))
A<-rbinom(n,size=1,PA.1givenWV)

## Generate random Y given A,W,V
beta0<--2
beta1<-1
beta2<--1
beta3<-0.5
beta4<-0.7
PY.1givenAWV<-1/(1+exp(-(beta0+beta1*W1+beta2*V+beta3*A+beta4*A*V)))
Y<-rbinom(n,size=1,PY.1givenAWV)
```

Making Substitution Estimator


```
> head(data.frame(W1,V,A,Y))

           W1 V A Y
1 -0.74076009 4 0 0
2  0.02393476 2 0 0
3  0.11965807 1 0 0
4  0.69188911 1 1 0
5 -0.29255064 0 0 0
```

Now we simply do the substitution estimator by:

1. Fit a model for Q_0, that is $Q_n(A, W)$ - *in this case we do simply logistic regression*
2. For an observation, get a prediction for $A = 1$ and $A = 0$, so $(Q_n(1, W), Q_n(0, W))$ and thus the difference of these two, say $Diff(W) = Q_n(1, W) - Q_n(0, W)$
3. Get the average of these differences, $Diff(W)$, for all groups separate for each $v = 1, 2, \ldots$, so $\theta_n(v) = E_n(Diff(W) \mid V = v)$.
4. Plot these $\theta_n(v)$ vs. v.

R code for Substitution Estimator Using Traditional Regression Estimators


```
dat=data.frame(W1,V,A,Y)
### Model fit for Y | A,W1,V
AV<-A*V
glm.YgivenAWV<-glm(Y~W1+V+A+AV,family=binomial,data=dat)
summary(glm.YgivenAWV)
```

```
Call:
glm(formula = Y ~ W1 + V + A + AV, family = binomial, data = dat)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.90378  -0.39486  -0.13500  -0.06411   2.95951

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.9282     0.4111  -4.690 2.73e-06 ***
W1            0.8883     0.4115   2.159 0.030887 *
V            -1.2457     0.3768  -3.306 0.000946 ***
A             0.3369     0.5356   0.629 0.529303
AV            1.2153     0.4361   2.787 0.005323 **
---
Signif. codes:  0 ?***? 0.001 ?**? 0.01 ?*? 0.05 ?.? 0.1 ? ? 1

    Null deviance: 243.17  on 499  degrees of freedom
Residual deviance: 182.06  on 495  degrees of freedom
AIC: 192.06
```

R code for Substitution Estimator

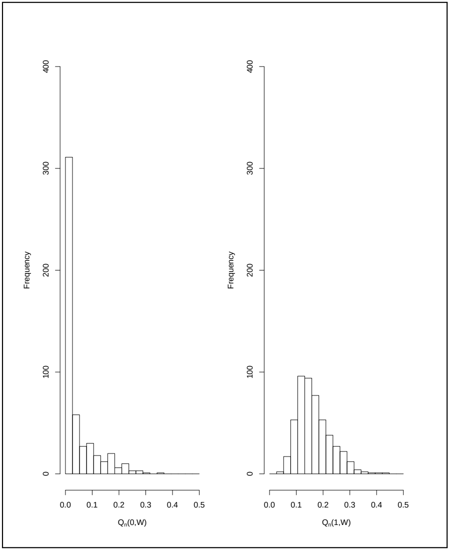
Get the predicted values for each observation at $A = 1$ and $A = 0$ ($Q_n(1, W), Q_n(0, W)$) and was their difference, $diff(W)$.

```
### Setting a = 0, and getting P(0,W)
p0.W<-predict.glm(glm.YgivenAWV,newdata=data.frame(W1=W1,V=V,A=rep(0,n),
   AV=rep(0,n)),type="response")

### Setting a = 1, and getting P(1,W)
p1.W<-predict.glm(glm.YgivenAWV,newdata=data.frame(W1=W1,V=V,A=rep(1,n),
   AV=V),type="response")
```

```
head(data.frame(Y,A,V,round(W1,4),predA0=round(p0.W,3),predA1=round(p1.W,3),diff=round(p0.W-p1.W
```

```
  Y A V      W1 predA0 predA1   diff
1 0 0 4 -0.7408  0.001  0.085 -0.085
2 0 0 2  0.0239  0.012  0.164 -0.152
3 0 0 1  0.1197  0.044  0.180 -0.136
4 0 1 1  0.6919  0.072  0.268 -0.196
5 0 0 0 -0.2926  0.101  0.136 -0.035
6 0 0 2 -0.6378  0.007  0.098 -0.091
```

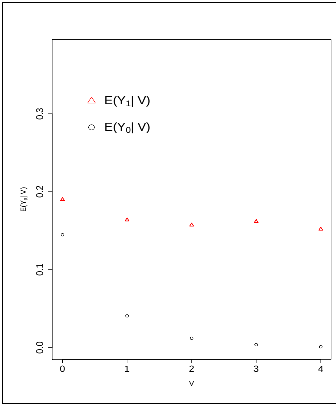
Distribution of Predicted Values at $A = 0$ and $A = 1$ ($Q_n(a, W)$)



R code for Substitution Estimator, cont.

Get estimates of the $E(Q_0(1, W) - Q_0(0, W) \mid V = v)$ for each of the $V = (0, 1, 2, 3, 4)$.

```
## Estimate E{E[Y|A=a,W]|V} with smooth regression
## First a=0 (just gets average of the predicted
##  values at each V=v.
EY0.WgivenV<-lm(p0.W~factor(V))
atV<-0:4
EY0.V<-predict(EY0.WgivenV,newdata=data.frame(V=atV))
## Then a=1
EY1.WgivenV<-lm(p1.W~factor(V))
EY1.V<-predict(EY1.WgivenV,newdata=data.frame(V=atV))
par(mfrow=c(1,1))
plot(atV,EY0.V,type="p",xlab="V",
      ylab=expression(paste("E(",Y[a],"| V)",sep="")),
  ylim=c(0,0.38),xlim=c(0,4),pch=1)
points(atV,EY1.V,pch=2,col=2,lwd=2)
legend(0.25,.25,
     c(expression(paste("E(",Y[1],"| V)",sep="")),
     expression(paste("E(",Y[0],"| V)",sep=""))),
     pch=c(2,1),col=c(2,1),bty="n")
```



Marginal Structural Models (MSM)

- Returning to simulated example, where the parameter of interest is E(Ya|V).

- As opposed to just getting empirical fit (connecting the averages),we now assume a model, or, for instance:

$$E(Y_a \mid V) = m(a, V; \beta) \overset{e.g.}{=} \beta_0 + \beta_1 a + \beta_2 V + \beta_3 a * V$$

  or if outcome is binary:

$$m(a, V; \beta) \overset{e.g.}{=} \frac{1}{1 + e^{-(\beta_0 + \beta_1 a + \beta_2 V + \beta_3 a * V)}}$$

- What one gains and loses is same as fitting a line through a set of averages as opposed to just reporting these averages:
  - → Gains power (reduction of variance of estimation) and simplicity of reporting by borrowing across the different estimates of $E(Y_a \mid V)$
  - → Loses by adding bias (typically $m(\cdot)$ is not the true model).

R code for Substitution based MSM


```
pred=c(p0.W,p1.W)
### Then,corresponding covariates in MSM
Vn=rep(V,2)
An=c(rep(0,n),rep(1,n))
datn=data.frame(Ystar=pred,V=Vn,A=An,AV=An*Vn)
glm.msm=glm(Ystar~A+V+AV,data=datn,family=binomial)
### Fit logit-linear model of prediction versus
### Get Results
summary(glm.msm)
## Function to get estimates and inference on
## exponentiated scale
## (e.g., get back OR's after logistic regression)
lreg.or <- function(glm.mod,robust=FALSE) {
    if(robust==TRUE){
      glm.1<-robcov(glm.mod)
      se=sqrt(diag(glm.1$var))
      cf=glm.1$coefficients
      lreg.coeffs=cbind(cf,se) }
     if(robust==FALSE) {
      lreg.coeffs <- coef(summary(glm.mod))}
     p=dim(lreg.coeffs)[1]
     l95ci <- exp(lreg.coeffs[2:p,1] -
       1.96 * lreg.coeffs[2:p ,2])
     or <- exp(lreg.coeffs[ 2:p,1])
     u95ci <- exp(lreg.coeffs[2:p ,1] +
    1.96 * lreg.coeffs[2:p ,2])
    pvalue=(2*(1-
       pnorm(abs(lreg.coeffs[,1]/lreg.coeffs[,2]))))[2:p]
    lreg.or <- cbind(l95ci, or, u95ci,pvalue)
    lreg.or
          }
lreg.or(glm.msm)
```

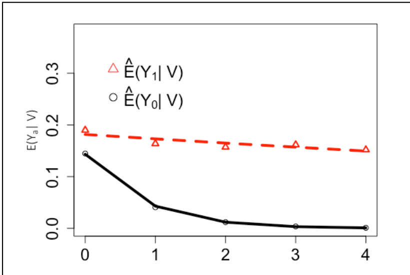
Results - ignore the inference (need to take another approach to get correct SE's).

```
> summary(glm.msm)

Call:
glm(formula = Ystar ~ A + V + AV, family = binomial, data =

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.38679  -0.08478  -0.00567   0.04823   0.59330

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.7877     0.2610  -6.850 7.36e-12 ***
A             0.2819     0.3280   0.859 0.390158
V            -1.3168     0.3191  -4.127 3.68e-05 ***
AV            1.2586     0.3303   3.810 0.000139 ***
---


> round(lreg.or(glm.msm),4)
    l95ci     or  u95ci pvalue
A  0.6970 1.3256 2.5213 0.3902
V  0.1434 0.2680 0.5009 0.0000
AV 1.8424 3.5203 6.7262 0.0001
```

R code for Displaying Results

```
*** Get contrasts
 glm.post.estimate=function(glmob,comps,
   exponentiate=TRUE) {
    if(is.matrix(comps)==FALSE) {
     comps=t(as.matrix(comps))}
 vc=vcov(glmob)
 ests = coef(glmob)
 linear.ests=as.vector(comps%*%ests)
 vcests=comps%*%vc%*%t(comps)
 ses=sqrt(diag(vcests))
    pvalue=(2*(1-pnorm(abs(linear.ests/ses))))
    if(exponentiate) {
      l95ci <- exp(linear.ests - 1.96 * ses)
      or <- exp(linear.ests)
      u95ci <- exp(linear.ests + 1.96 * ses)
      summ <- cbind(l95ci, or, u95ci,pvalue)
      }
            if(exponentiate==FALSE) {
      l95ci <- (linear.ests - 1.96 * ses)
      logor <- (linear.ests)
      u95ci <- (linear.ests + 1.96 * ses)
      summ <- cbind(l95ci, logor, u95ci,pvalue)
      }
      return(summ) }

###  Get OR of E(Y(1)|V=0) vs. E(Y(0)|V=0) and
###    E(Y(1)|V=2) vs. E(Y(0)|V=2)

comps=rbind(c(0,1,0,0),
            c(0,1,0,2))

rownames(comps)=c("Caus.OR: V=0","Caus.OR: V=2")
```

```
> round(glm.post.estimate(glm.msm,comps),4)
                  l95ci      or    u95ci pvalue
Caus.OR: V=0 0.6970  1.3256  2.5213 0.3902
Caus.OR: V=2 5.3365 16.4278 50.5714 0.0000

## Plot Results
predV.0=predict(glm.msm,newdata=
data.frame(A=rep(0,5),V=0:4,AV=rep(0,5)),
type="response")
lines(0:4,predV.0,lwd=4)
predV.1=predict(glm.msm,newdata=
data.frame(A=rep(1,5),V=0:4,AV=0:4),
type="response")
lines(0:4,predV.1,lty=2,col=2,lwd=4)
```



Use the tangibility and form of substitution estimators to inspire other parameters

David McCoy PhD, MSc
Division of Biostatistics, UC Berkeley
Edward's Lifesciences

- Other types of parameters can be motivated and estimated using substitution methods.
- Consider counterfactual rules, not simply treatment levels.
- For example, if we have the same model as we just discussed $V \subset W \to A \to Y$.
- Consider a counterfactual based on the application of a rule of the form $d(V)$, where one wants the mean if everyone were given treatment according to $A = I(d(V) < v)$, that is, for everyone younger than a particular age v.
- Often time, these counterfactual scenarios are much more practical than giving everyone the same treatment, particularly in biomedical applications.

Treatment rule impacts from substitution estimator, cont.

- One might be interested in the causal parameter $EY_{d(v;V)}$, where $d(v;V) = I(V < v)$, so $= 1$ (e.g. give tx) if age is less than v, but 0 (no tx) otherwise.
- One could estimate this parameter for different values of $V = v$ and plot them to find an optimal v to use.

Practice problem

MOVE INTO THE RMD FILE FOR PRACTICE

- Consider the data on canvas called Rule.csv, which has the data of $V \subset W \rightarrow A \rightarrow Y$ generated according to same causal model as above.
- Estimate the regression model $Q(A, W) = E(Y \mid A, W)$ using lm in R and the following functional form: $Q(A, W) = b_0 + b_1 A + b_2 V + b_3 W_1 + b_4 A * V + b_5 A * V^2$.
- Use the fit to estimate $E(Y_{d(v;V)})$ for $v = 0, 1, 2, 3, 4$.
- Plot $\hat{E}(Y_{d(v;V)})$ vs. v.

Summary

- Same method can be used for many parameters related to theoretical interventions at one time (point treatment cases), including:
  - → treatment rules,
  - → mediation impacts (direct and indirect effects),
  - → stochastically assigned interventions,
  - → etc.
- Substitution estimators are "relatively" intuitive.
- When estimating the regressions with parameteric models, can use the (nonparametric) bootstrap to get inference (or the delta-method).
- Though these estimators work when machine learning is used, the don't result in an estimator with predictable sampling distribution.

References

J.M. Robins.
Association, causation, and marginal structural models.
Synthese, 121:151–179, 1999.